Indoor Marker-based Localization Using Coded Seamless Pattern for Interior Decoration

Shigeru Saito[1]
Graduate School of Information Science and Technology, The University of Tokyo.

Atsushi Hiyama[2]
Graduate School of Information Science and Technology, The University of Tokyo.

Tomohiro Tanikawa[3]
Graduate School of Information Science and Technology, The University of Tokyo.

Michitaka Hirose[4]
Graduate School of Information Science and Technology, The University of Tokyo.

ABSTRACT

Because marker-based position tracking system is inexpensive and easy to use, it has the potential to make the system more feasible for homes or businesses and broaden the current suite of augmented reality (AR) techniques. To apply marker-based systems to homes or businesses, blending markers with environments naturally is important. Our innovative approach to marker-based 3D position tracking uses seamless patterns encrypted with positional data. Although users can obtain 3D positional data by processing marker images similar to many existing marker-based systems, our markers are designed with interior decoration in mind. That way they can be installed in walls, floors, or ceilings. Unlike existing systems whose fiducial markers were designed first and foremost to be processed by computers, ours are visually attractive. By integrating positional information within the interior design, our system enables users to enjoy the benefits of position tracking without being constantly aware of the system's presence.

We developed a method for making patterns in which positional information is encrypted and a method for calculating the 3D position of a user by decoding those patterns. We then constructed a system using three patterns that were made by the proposed method and evaluated that system.

CR Categories: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques; I.4.0 [Image Processing and Computer vision]: General—Image Processing Software;

Keywords: Indoor Position Tacking, Interior Decoration, Coded Pattern, Fiducial Marker, AR

1 INTRODUCTION

Due to the development of augmented reality (AR) technologies such as navigation annotation displays with portable terminals and wearable computers, considerable research has recently been done on the best techniques for presenting information based on a user's position [1]. For example, Malaka, et al., Daehne, et al., and Tenmoku, et al. proposed AR navigation systems that utilize wearable computers ([2], [3], [4]).

In the ideal location-based information system, users should be able to obtain accurate positional data in real time. Furthermore, these systems should work well over large areas, be easy to setup, and have a low cost. For pinpointing people's locations in outdoor environments, the global positioning system (GPS) are common and have already been used in a variety of regions. However, since GPS does not work inside buildings, many alternative indoor tracking systems have been proposed but each comes with its own set of problems.

One kind of indoor positioning systems uses transmitters and receivers capable of picking up infrared, magnetic, or other wireless communications [5][6]. Although these systems are highly accurate, their range is limited and the components are expensive. Furthermore, the set up and calibration of such devices is not trivial; a big, heavy transmitter or multiple smaller ones have to be installed and calibrated, and the users have to equip themselves with sensors, amplifiers and power sources.

To avoid these constraints, other systems use cameras and image processing. There are two basic image processing techniques used in these types of systems. One analyzes an environment's scenery, matching the result with captured images [7-13], and the other uses fiducial markers placed in the environment [15-19]. Although camera systems are lower in cost and have broader coverage than wireless systems, there are problems. Camera systems that analyze the scenery do not respond well to scene changes and in the versions that use markers, those markers stand out, reducing the aesthetic appeal of a room and distracting the user. Although there is also research on hidden markers to increase the aesthetic appeal of the system, users have to be equipped with proliferation of devices such as infrared camera and infrared light simultaneously [20]. Therefore, the current state of the art has not advanced beyond research and these systems are not found in homes, museums, shopping malls, or offices.

After determining the limitations of the current systems, we have identified four constrains that, if met, will broaden the appeal AR information presentation and tracking systems. They must be, easy to use, inexpensive, aesthetically appealing, and effective in any size space. Since the ubiquity of the digital camera has made systems with fiducial markers approachable and economic, we proposed a marker-based localization system, which can be incorporated into the interior décor of buildings. In order to give our markers visual appeal, we propose the system that embeds positional data into seamless patterns that can be printed on wallpaper or floor tile. By blending positional information into the interior décor, it is possible to have a positioning system that enables users to enjoy the benefits of a tracking system without being constantly reminded of its presence.

2 RELATED WORK

A great deal of research has been done on vision-based localization systems and, as noted above, two prominent types have emerged.

In their experiments with scene analysis, Kourogi, et al. matched current images from video frames with database recordings of the same scene in order to determine a user's location [7]. Sticker, et al. and Lee, et al. calculated positional information using a 2D affine transformation [8][9], and Vacchetti, et al. used edge and texture information to compare reference

[1]e-mail: saito@cyber.rcast.u-tokyo.ac.jp
[2]e-mail: atsushi@cyber.rcast.u-tokyo.ac.jp
[3]e-mail: tani@cyber.rcast.u-tokyo.ac.jp
[4]e-mail: hirose@cyber.rcast.u-tokyo.ac.jp

IEEE Virtual Reality Conference 2007
March 10 - 14, Charlotte, North Carolina, USA
1-4244-0906-3/07/$20.00 ©2007 IEEE

images to captured images [10]. Davison, et al. presented a top-down Bayesian framework for single camera localization by mapping a sparse set of natural features, using motion modeling and an information-guided active measurement strategy [11]. Behringer, et al. matched images with models of the surroundings [12], while Simon, et al. proposed an efficient solution to real-time camera tracking for scenes that contain planer structures [13].

Although some of these solutions were designed for outdoor augmented reality, they are adaptable for indoor localization. The advantage of these systems is that they don't have to have any beacons that might not fit inside a room. A limitation of these systems is that if the environment changes, the system loses its point of reference, and the tracking will fail. For this reason, other researchers have experimented with fiducial markers, which can also provide positional data.

Some systems use markers that are based on ARToolkit developed by Kato, et al. [14]. ARToolkit markers have a square frame, and by taking advantage of two opposing edges of the frame, a camera's position can be calculated. Thomas, et al., Kalkusch, et al. and Baratoff, et al. [15-17] have developed localization systems using ARToolkit markers. In these systems, the researchers used the patterns in the marker frames to calculate markers' IDs. However, the number of markers was limited to no more than one thousand. To get around this constraint, Naimark, et al. developed original circular markers that offered much higher information density, more than ten thousand markers in all [18]. As for the rest, Rao, et al. proposed a system for indoor immersive environments using video projectors to display a grid pattern on the floor [19]. The problem with these markers or patterns was that their frames or dots were designed with only the requirements of the computer processing in mind. The art of their form was brushed aside in favor of function and there was no flexibility in the design to allow humans to create patterns they would find engaging.

To address this concern, researchers would have to design markers that contained positional data, and at the same time, took aesthetic design into consideration. Nakazato, et al. found a way to avoid the problem entirely by developing invisible retro-reflective markers. The placement of these markers could be then be captured with an infrared camera and an infrared light to obtain the user's 3D position [20]. Unfortunately, the system requires specialized devices such as infrared cameras, which most users would have to purchase and set up solely for the purpose of localization.

Another solution that was considered was an expansion of the Anoto pattern—a pattern of minute dots printed on special paper. Used in conjunction with the Anoto digital pen, which contains a camera, this paper makes it possible to calculate the trajectory of a person's writing [21]. Clusters of 36 dots indicate positional data and they are so small that it is difficult for humans to see them. Although Anoto patterns express an enormous number of markers, enough to cover sixty million square kilometers, this system is only useful in paper or notebooks. To adapt it for an indoor tracking system, the dots must be enlarged so the camera can capture them, but such large dots would clash with most interiors.

3 Proposed System

Our proposed system, illustrated in Figure 1, introduces a way to make patterns that could be incorporated in any interior design, in the form of wallpapers, carpets, or floor tiles. These patterns consist of clusters of markers and are optimized for visual pleasure with the aim to unobtrusively integrate them into homes and businesses. As with most of the other systems that used

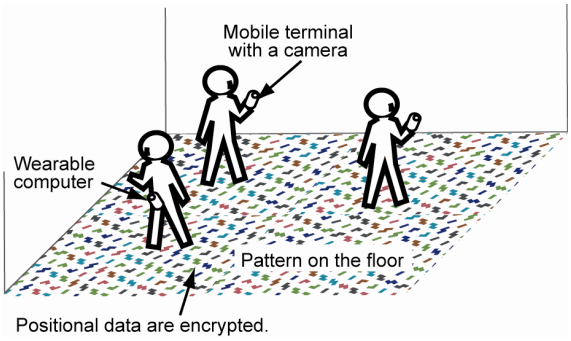


Figure 1: Proposed system.

fiducial markers, our users can obtain their positional data in real time from these patterns using the kinds of USB cameras found in most electronics stores.

Our system has the following advantages over previous systems:

- Hidden markers

Markers that reveal positional information are hidden in the patterns. By regulating certain parameters of each figure within these patterns, we could encode them with positional information. In particular, the ID number of each marker is determined by the rotational angles of figures, and the camera's orientation toward each marker is obtained from positions of figures. Many preexisting decorative patterns can be easily rearranged so that computers can process them to pinpoint locations in 3D.

- Flexibility of design

Our marker-encoded patterns can be generated using a single algorithm presented later in this paper. By calculating user's position using contour information and positions of our figures, we avoid the limitations of fiducial markers that have to use base figures such as dots and frames. Avoiding base figures enables us to make many types of patterns.

- Broad coverage

By using the rotational angles of the figures in the patterns to express marker ID numbers, a single figure can express more than ten IDs. Therefore, simply adding more figures increases the number of IDs exponentially. Consequently, if we ignore the aesthetic constraints and apply our method to other preexisting markers, which can express more than a thousand ID numbers, we can express a sufficient number of markers to cover almost any sized room.

Our system also has the same advantages as other marker-based systems. It is relatively inexpensive to print these markers instead of deploying custom devices in the environment. Furthermore, it is simpler to use because the only electronic device required is a digital camera. In addition to these benefits, multiple people can use our system simultaneously.

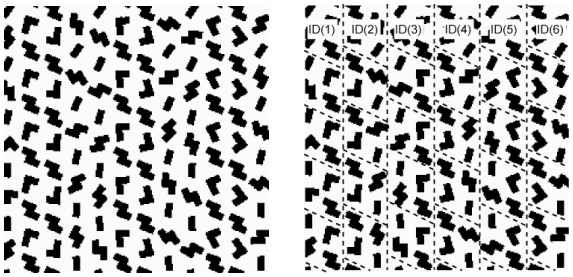
4 Method of Making Patterns

We invented a method for making coded patterns. This section explains which types of pattern we chose to embed with positional information and why, how to encode that information, and how to make alternative patterns.

4.1 Pattern Type Our System Target

The patterns shown in Figure 2 are commonly used for walls or floors [22][23]. In these patterns, the figures of leaves and balloons, while ordered, are not arranged in a way that would be meaningful to a computer trying to calculate positional data. It is

Figure 2: Example of patterns in wallpaper or floor tile [22][23].



Pattern (a)

(i) (ii)

Figure 3: (i) Coded patterns we made as the first step. (ii) Arrangement of markers.

possible to encrypt these patterns with computationally meaningful information by regulating certain parameters such as the rotational angle and positions of the figures.

According to this idea, as a first step, we made the information-encoded pattern as shown in Figure 3(i). We started with Pattern (a) (see Figure 3) because it is easier for the computer to identify each shape by image processing because each figures have corners that are easy for the camera to detect. More visually aesthetic patterns, in which shapes of figures are more difficult to identify, will be described in Section 6. In the next section, we describe the basic algorithm for encoding and decoding Pattern (a).

4.2 Marker

4.2.1 Mechanism of Marker

Pattern (a) contains four types of figures, which seem to be arranged randomly, however, they are not. Their order is regulated by the parameters of rotational angles and barycentric positions, from which users can obtain their positional data. Markers that have ID numbers related to coordinate values in the plane of the pattern are arrayed within the pattern. Figure 3 (ii) illustrates how these markers are arranged.

In order to obtain 3D positional information, it is necessary to recognize a markers' ID number and calculate the pose of each marker. In the proposed method, the rotational angles of the four types of figures express one marker's ID number and the barycentric positions of the figures enable us to calculate each camera's orientation. Figure 4 illustrates an arrangement of four figures that makes up one marker.

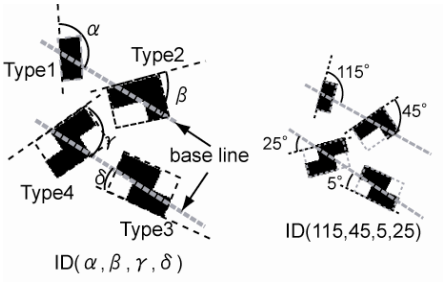


Figure 4: A marker and each figure's specification

We call each figure in Pattern (a) Type 1, Type 2, Type 3 and Type 4. Type 3 and Type 4 share line symmetry. Rotational angles are calculated by the angular difference between the base lines, which are the grey dotted lines shown in Figure 4, and the longer side of the minimum rectangle that encloses each figure. Base lines are the ones that connect the barycentric positions of two figures. In this case, they are the lines that connect Type 1 to Type 2, and Type 3 to Type 4. These two lines are parallel in this pattern but they do not have to be. Since shapes such as a rhombus have minimum rectangles that can be calculated in more than one way, they are not useful for this method. The rotational angles for shapes with properties like those of a rhombus are calculated by designating one of the sides or a chief axis as the angle vector.

The barycentric positions of figures in Pattern (a) are arranged in such a way that they are the vertices of a parallelogram whose inner angles are 60 degrees and 120 degrees and each figure is 64 mm from its nearest neighbor. The image processing algorithm on the user's system can find the orientation of camera and provide the user's 3D positional data using this information.

4.2.2 The Number of Markers

This method enables a few types of figures to express many different markers, because it uses the rotational angles of the figures. Therefore, it is necessary to know the smallest angular difference the camera can distinguish in order to systematically make the markers. Consequently, we conducted an experiment to determine the camera's ability to recognize the angular differences between each figure.

We prepared markers in which figures were rotated 5, 10, 15, and 20 degrees, and measured the recognition error rate under three different conditions; we set up all the cameras at a distance of 1000 mm and had one with its lens perpendicular to the marker, another with its lens tilted 45 degrees, and a third that was rotated freely around the marker. Distinctions were made among the markers by using the intermediate value of the angle in each marker. The recognition error rate was calculated as the number of markers that were recognized incorrectly out of the number of all recognized markers in the captured images. Although in this experiment we set the camera 1000 mm away from the pattern, we will discuss the relationship between camera distance from Pattern (a) and the recognition error in later section.

The result is indicated in Table 1. Since the captured images blurred when the camera moved, values of the error rate were

Table 1: Angular difference between each marker and recognition error rate.

Difference of Angle [degrees]	Recognition Error Rate [percent]		
	0 [degrees]	45[degrees]	Moving
5	0.9	14.0	44.9
10	0.5	1.1	7.2
15	0	0.1	1.9
20	0	0.1	1.2

bigger than with the other conditions. Under all the conditions, the recognition error rate decreased when difference of the angle was 10 degrees or greater. In order to be able to express as many markers as possible while still keeping the error rate relatively low, we decided to use 10 degrees as the angular difference even though the recognition error rate was 7.2 percent under the condition that the camera moved. After making this decision, we laid out the pattern in such a way that each type was rotated with 10 degrees angular differences from each other at 5, 15, 25, …, 175 degrees. Since the process uses rectangles that become symmetrical at 180 degrees, each figure can express 18 different markers. So, in total Pattern (a) can express a total of 104,976 (18^4) markers.

However, on the other hand, an error detecting algorithm must be taken into consideration because it is necessary to detect errors and further reduce the recognition error. Therefore, we reserved one figure (Type 4 in the case of Pattern (a)) in every marker to be used for error detection. The rotational angle of error detecting figure is determined in advance and each marker is recognized only when the angle of error detecting figure calculated from images corresponds with this predetermined angle. With one figure reserved as error detection purpose, Pattern (a) can express a total of 5,832 (18^3) markers. If one marker is printed so as to take a 100 mm x 100 mm area, Pattern (a) can cover a 58 m^2 of floor space. More space can be covered by separating markers with gaps or other designs that are not related to the marker calculations. Furthermore, the number of markers can easily be increased by adding more types of figures. For example, we made a pattern that consists of six figures and therefore it can express 18^5 (1,889,568) markers. This pattern will be described in more detail in Section 6. What this exponential increase indicates is that our method allows a few figures to express a vast number of markers.

4.3 Arrangement of Markers

In pattern (a), Type1, 2 and 3 indicate ID numbers and Type 4 works as the error detecting figure. So, in this case we expressed the ID number as "(α, β, γ)," where the letters α, β, and γ would be the rotational angles of Type 1, 2, and 3 respectively. In the patterns we discuss in this paper, we arrayed the markers in numerical order. In particular, the markers were arranged in such a way that the value of α changed along one of coordinate axis and that of β changed along the other axis as illustrated in Figure 5 (i). Although in Figure 5 (i), the value of γ is constantly 5, it is possible to cover up to 18 times the area by arranging areas where values of γ are different. In the area in which γ is constant, by arranging the patterns this way, the central position of each marker, (x_α, y_β), can be calculated with one simple equation as Equation (1) demonstrates. In this equation, the central point of (5, 5, γ) marker is set as the origin. Letter D in Equation (1) is the distance between markers.

$$(x_\alpha, y_\beta) = (\frac{(\alpha - 5) \times D}{10}, \frac{(\beta - 5) \times D}{10}) \quad (1)$$

To aid with the image processing, each type of figure must be arranged clockwise starting with Type 1 and ending with Type 4, and all the figures must be D/2 away from each other as required by the image processing algorithm. So if all the markers are arranged in the same direction, marker recognition is conducted by drawing dotted and solid circles around each group (Figure 5 (ii)). Those markers which have odd values of (β-5)/10 are rotated 180 degrees, because it is necessary to avoid marker arrangements that would allow recognition to be conducted in more ways than one.

By using this method described in this section, we can make many types of patterns by changing the shapes and positions of figures, the size of each figure, or the number of types of figures.

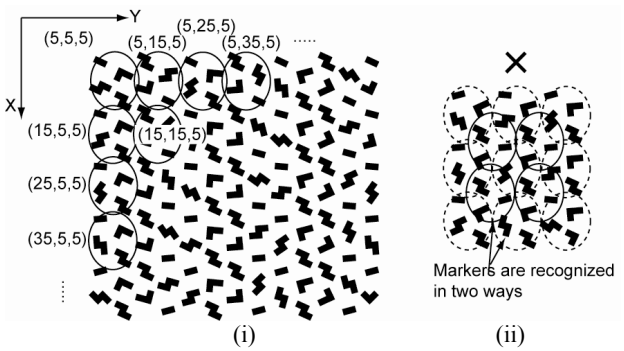


Figure 5: (i) Arrangement of markers.
(ii) Arrangement of markers that are recognized in two ways

5 Method of Decoding Patterns

Users can get 3D positional data by processing captured images of patterns made by the method proposed in Section 4.

5.1 Image Processing Algorithm

The processing flow is illustrated in Figure 6. First, the images are binarized and all the black areas (the figures) are labeled. After labeling, each figure is categorized by type. In the case of Pattern (a), information about the number of corners on each of the figures and their positional relationships is used in this sort. Then a set of each type is grouped as one marker. A group is recognized by the distance between each type's figure and the alignment order. As described in 4.3, all types of figures in one marker must be a certain distance apart and ordered clockwise from Type 1 to Type 4.

The orientation of the camera, calculated using the affine transformation matrix, is obtained from the barycentric positions of the figures in these groups. Since the positional relationships of these figures are given in advance, comparing them with captured images of four matching figures, allows the orientation of camera to be calculated. By using the affine transformation matrix, to change the contour coordinates of the figures in the image's plane coordinate system to the world coordinate system, the world coordinates of the minimum rectangle of each figure can be calculated. These world coordinates are necessary for calculating markers. After the rectangles are described, IDs of markers are determined by the process detailed in Section 4.1.

Finally users can get 3D positional data from the markers after an application of the affine transformation matrix (see Section

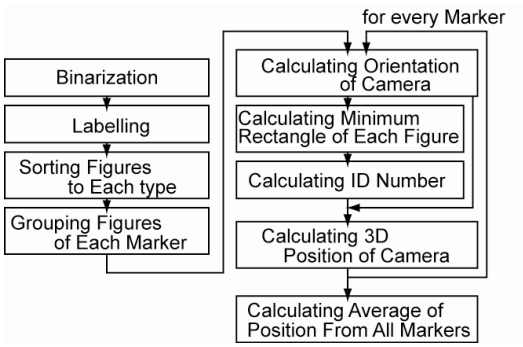


Figure 6: Program flow.

5.2). In the event that several markers are recognized in one frame, calculations of the affine transformation matrix and positional data from all markers are used, and the average of all that positional data is adopted as the real world position in that frame.

5.2 Calculation of Position from One Marker

At this stage, we are ready to calculate the real world position of a user from a single marker. For this calculation, the four coordinate systems of the camera image plane, the camera itself, the marker, and the world, and three transformation matrices, P, T_{cm} and T_{mw}, are used. The relationship between each coordinate system is shown in Figure 7.

The perspective projection matrix P, which is obtained by camera calibration, is calculated in advance by an algorithm which Kato, et al. proposed [12]. The relation between the image plane coordinates (X_i, Y_i) and the camera coordinates (X_c, Y_c, Z_c) are described by matrix P in equation (2).

$$\begin{pmatrix} hX_i \\ hY_i \\ h \\ 1 \end{pmatrix} = P \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix}, \quad P = \begin{bmatrix} P_{11} & P_{12} & P_{13} & 0 \\ 0 & P_{22} & P_{23} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

T_{cm} indicates the transformation matrix from the camera coordinate system to the marker coordinate system, and T_{mw} indicates the transformation matrix from the marker coordinate system to the world coordinate system. The T_{mw} of each marker is obtained in advance from the method of marker arrangement discussed in Section 4.3. The relationship between these transformation matrices and the coordinates is calculated using the expression of homogeneous coordinates in Equation (3).

$$\begin{pmatrix} hX_i \\ hY_i \\ h \\ 1 \end{pmatrix} = P \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = PT_{cm} \begin{pmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{pmatrix} = PT_{cm}T_{mw} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

$$T_{cm} = \begin{bmatrix} R_{cm3\times3} & T_{cm3\times1} \\ 0\ 0\ 0 & 1 \end{bmatrix} \quad T_{mw} = \begin{bmatrix} R_{mw3\times3} & T_{mw3\times1} \\ 0\ 0\ 0 & 1 \end{bmatrix} \tag{3}$$

In Equation (3), PT_{cm}, which is given in image plane coordinates, is actually determined by the positional relationship of the four figures in a captured image, using an algorithm proposed by Kato, et al. [12]. Since P is given, T_{cm} is obtained by $P^{-1}PT_{cm}$.

Then Equation (3) is converted to Equation (4).

$$\begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} = T_{mw}^{-1} \begin{pmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{pmatrix} = T_{mw}^{-1} T_{cm}^{-1} \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} \tag{4}$$

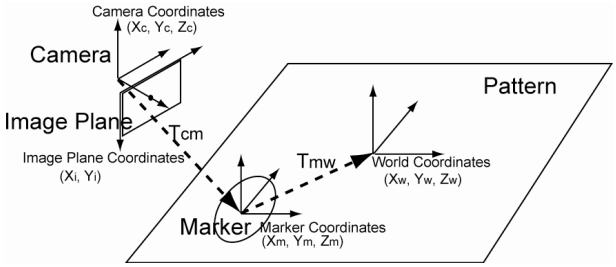


Figure 7: Relations of coordinate systems.

When T_{mw}^{-1} and T_{cm}^{-1} are calculated, the coordinates of the camera, which are expressed in world coordinates (X_w, Y_w, Z_w) can be obtained by substituting $(X_c, Y_c, Z_c) = (0, 0, 0)$ for Equation (4). Thus we can get camera's 3D position from each marker.

5.3 Implementation

We implemented the algorithm using a 2.16GHz SONY VAIO laptop PC with an Intel Core Duo processor and a CREATIVE Live! Pro USB camera (VGA, 30fps) and programmed the application shown in Figure 8. The upper part of the window in Figure 8 is a view of the marker recognition, and the window below it displays of the camera's position as drawn by 3D CG.

We used the ARToolkit library for binarization, labeling, and calculating the affine transformation matrices, and the OpenCV library for calculating the minimum rectangle enclosing each figure.

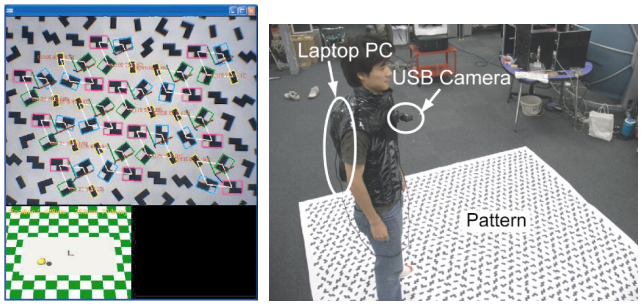


Figure 8: Implementation.

6 Extension to Other Patterns

While we have described the method we used to make and decode Pattern (a), the figures in that particular sample pattern were identified by the number of corners they had as well as their positional relationships. That method can only be used for patterns with figures that have corners with simple contours, and of course, cornerless figures or ones with complicated contours are used in decoration all the time. Therefore, we extended the system so that it would work for these patterns.

We made the two sample patterns shown in Figure 9. Pattern (b) on the left is made up of four types of leaves that have complicated contours and Pattern (c) on the right is made up of six cornerless figures. Pattern (b) has 18^3 markers, while Pattern (c) has 18^5 markers since one figure in each pattern serves as an error detecting figure. These patterns are visually aesthetic and suitable for carpets or wallpapers in indoor environments. However, in order for these patterns to work with our system, each of their figures has to be distinguishable by image processing.

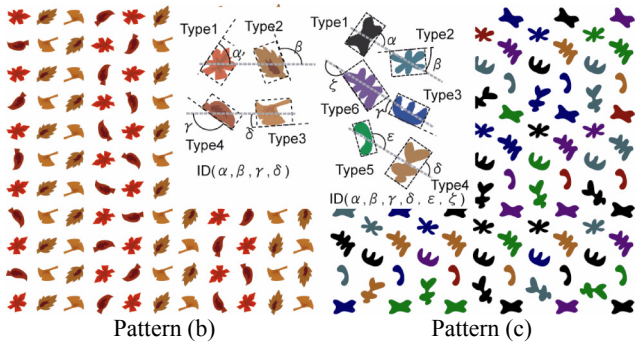


Figure 9: Examples of patterns.

6.1 Discernment by P-type Fourier Descriptor

To discern the different figures in these patterns, we used a P-type Fourier descriptor [24]. The P-type Fourier descriptor is usually used as a descriptive method for a linear drawing using the Fourier transformation. The vector between some points on a linear drawing are Fourier transformed. These transformed values consist of real parts and imaginary parts and are invariant for displacement and scaling, Furthermore, their absolute values are invariant for rotation. In our system, the coordinate values of a figure's contour are available for the transformation. Since our system also uses contour information in calculating the minimum rectangles of figures, it increases efficiency to use that same information in separate processes. An additional benefit is that our calculation times are faster than those of other algorithms that use raster scanning, because only the contour information is required.

Thus, we decided to use the spectrum of absolute values of the P-type Fourier descriptor because of its invariant character for displacement, scaling and rotation, and for its processing efficiency.

6.2 Implementation

Any computer can calculate the P-type Fourier descriptor, using the Fast Fourier Transformation (FFT). However, the spectrums of descriptors in each figure range discretely depending on the type and size of the figures. Therefore, the maximum spectrum values need to be normalized in such a way that they are equal across all figures. Furthermore, if the captured images of the figures are small, the values slant to a high-frequency component. Therefore, the frequency also needs to be normalized by the number of pixels in the contour. Figure 10 illustrates 30 normalized values for all the figure types in Pattern (b). In order to actually distinguish the figures, the P-type Fourier descriptor of a sample of each figure type must be registered in advance, and the correlation factor between it and the descriptor of each labelled figure in the captured images must be used. Although each figure looks similar at a rough estimate, the grey area shown in Figure 10 shows the frequency area in which the spectrum values of each figure differ the most. By looking in this area, our algorithm is able to differentiate between each figure.

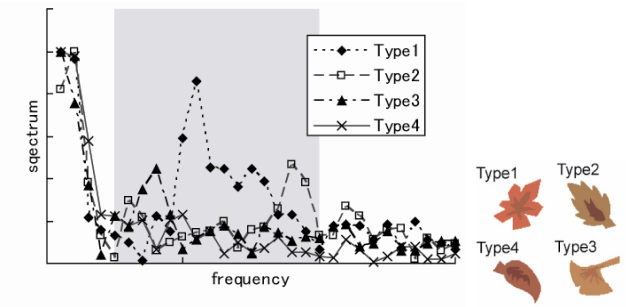


Figure 10: Spectrum of P-type Fourier descriptor of Pattern (b). (The grey area is used for discernment.)

Figure 11: Implementation.

Using the same laptop PC and USB camera as in Section 5.3, we implemented the application for two figures (Figure 8). Figure 11 shows the views of the marker recognition and the calculation results for the two patterns. We used the fftw library [25] for calculating P-type Fourier descriptors.

7 Evaluation

We evaluated the proposed system by experiments. We examined the possibility of the adaptive flexibility of the system for many patterns by indicating that users can get positional data with a certain level of recognition rate and accuracy using Patterns (a), (b), and (c). Furthermore, we measured the calculation speed to confirm that our system can run in real time.

7.1 Flexibility for Many Patterns

7.1.1 Different Patterns in Figures' Shape

To test the effectiveness of the system for each pattern, we printed the three patterns (Pattern (a), (b), and (c)) on papers and placed them on the floor. The three patterns are shown in Figure 3 and Figure 8.

Pattern (a) consists of four types of angulations arranged in such a way that they are closely packed together with a mere 64 mm of space between each one. Pattern (b) consists of four types of figures that have complicated contours, but the leaf motif is more meaningful to humans than the stark geometry of Pattern (a). These leaves are arranged in such a way that their barycentric positions are the vertices of squares in which the length of the sides is that familiar value of 64 mm. Pattern (c) is made up of six types of figures whose contours are curved and cornerless. As with Pattern (a) the figures of Pattern (c) are closely packed, 64 mm from one another.

To affirm that the rates will change in response to both the camera's tilt and the distance between the camera and the pattern,

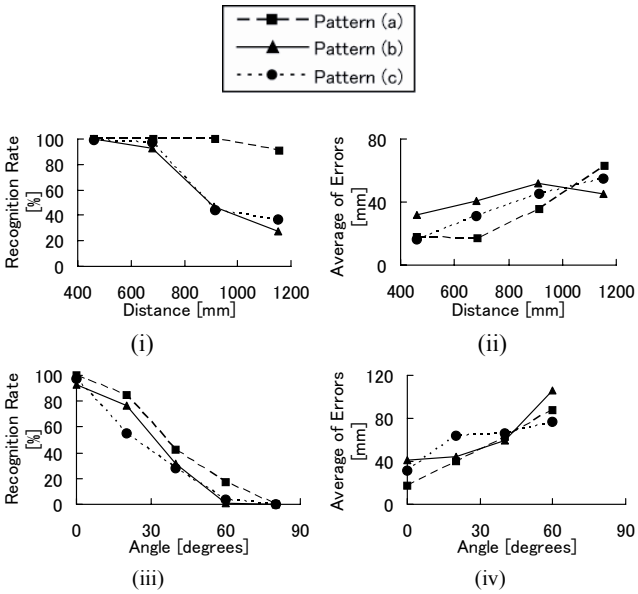


Figure 12: Results of the experiment in section 7.1.1.
Relation of (i) distance and recognition rate
(ii) distance and average value of position errors
(iii) camera angle and recognition rate
(iv) camera angle and average value of position errors.

measurements were taken at four different camera distances while the lens remained perpendicular to the pattern, and at five different lens angles while the camera was kept a consistent 680 mm away. In this experiment, the recognition rate is calculated as the rate of the number of recognized markers out of the total number of all markers in the captured image.

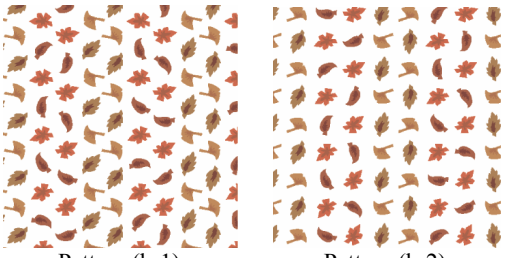
The results are indicated in Figure 12. From the results, we can see that the farther the camera is from the patterns and the more the camera tilts, the lower the recognition rate is and the larger the average value of the errors is in all patterns. Overall, there are some differences in the recognition rate between the three patterns, but there are not many differences in the averages of the errors.

From Figures 11 (i) and (ii), it is clear that Pattern (a) shows a higher rate of accurate recognition, even at the farthest distance than Patterns (b) and (c) did. This result suggests that, due to the distance of the camera and the resulting size reduction of the patterns in the captured images, it becomes more difficult to distinguish figures by the discernment method using P-type Fourier descriptors.

On the other hand, when it comes to camera angle, the recognition rate for all of the patterns decreases and the average value of the errors increases with any level of camera tilt. We believe this is due to larger error values in the affine transformation matrices as well as the difficulty of extracting the figures' feature points or determining good P-type Fourier descriptors. Errors in the transformation matrices would cause mismatches between calculated ID numbers and rotational angles of error-detecting figures at the ID number calculation stage.

7.1.2 Different Patterns in Sizes or Arrangements of Figures

To test the effectiveness of patterns that are different in size and arrangement, we printed the three patterns shown below (Figure 13). Pattern (b-1) is made from four types of leaves that are the same as Pattern (b). Each leaf is 64 mm away from its nearest neighbor in a closely packed arrangement. The placement of the barycentric positions of the four leaves situates them at the vertices of a parallelogram with inner angles of 60 degrees and 120 degrees. Pattern (b-2) is the same as Pattern (b) with the four leaves laid out to serve as the vertices of a square. And Pattern (b-3) is printed in such a way that the size is four times as large as Pattern (b-1) and so the figures are 128 mm apart. For these three patterns we measured the recognition rate and the average error



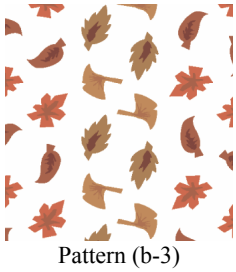


Figure 13: Patterns used in experiment in section 7.1.2.

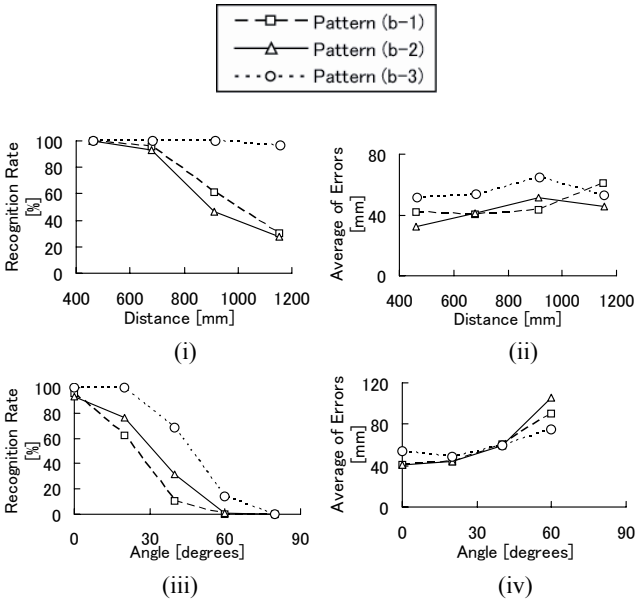


Figure 14: Results of the experiment in section 7.1.2.
Relation of (i) distance and recognition rate
(ii) distance and average value of position errors
(iii) camera angle and recognition rate
(iv) camera angle and average value of position errors.

value, applying the same experimental set up and methodology described in Section 7.1.1.

The results are indicated in Figure 14. As with the experiment in Section 7.1.1, the recognition rate decreases while the errors increase with camera distance and tilt. The experimental results show that the different arrangements in Pattern (b-1) and Pattern (b-2), cause little difference in both recognition and error rate. On the other hand, when comparing Pattern (b-1) with Pattern (b-3), we noticed that although Pattern (b-3) showed as high a recognition rate as (b-1), it produced slightly more errors. This is because since the ratio of pattern size and the error calculated for each marker is constant, the image processing algorithm registers more errors on the larger prints.

7.1.3 Discussion

From the results of the experiments described in Section 7.1.1 and 7.1.2, it can be expected that regardless of the type of pattern, users can obtain 3D positional data with as many recognition rate and accuracy as indicated in Figures 11 (b) and (d) or Figures 13 (b) and (d).

However, compared to existing systems that use fiducial markers, one of the problems with our system is the degradation in performance when the camera angle is larger than 30 degrees. This result is largely due to the increased error values in the transformation matrices. Although our markers are provided as wallpapers, floor tiles, or carpets that would cover large areas of a room, increasing the possibility that they would be captured by a camera that is nearly perpendicular to the plane of the pattern, the image processing algorithm should be robust enough to handle a wide range of camera angles. Using not only the positional relationships of the figures in one marker, but also the positional relationships of all the figures' arrangements might solve this problem. In addition, as the distance between the camera and the pattern as well as camera angle increases, it would be better if the P-type Fourier descriptor method had a higher marker recognition rate. By registering more of the figures' Fourier descriptors and using learning algorithms, we could probably increase the number

of correctly distinguished figures.

Another consideration is the effect of camera settings, such as the resolution or zoom, on the accuracy of this system. Changing the zoom of the lens or using a higher-resolution CCD would enable the camera to capture clearer pattern images, consequently increasing the accuracy of any calculations performed on pictures taken from far away.

7.2 Calculation Speed

This system must operate in real time. We measured how quickly our system could calculate position from markers found in Pattern (a), (b) and (c). If many markers are captured in one image, the average positional values of all the markers are adopted. So it is conceivable that the number of calculations, and therefore the time it takes to perform them, would increase in response to an increase in the number of recognized markers. To test this hypothesis, we took measurements of the calculation speed at a variety of recognition rates.

The results are shown in Figure 15. The more markers recognized, the fewer frames the computer could process in a given time period. Additionally, recognition speed of Pattern (a) was higher than that of the patterns that required P-type Fourier descriptors for recognition: that is because it is trivial to recognize figures using the number of corners. Introducing P-type Fourier descriptors results in more calculations. Furthermore, it requires more calculation time to process Pattern (c) than Pattern (b). This result suggests that the presence of additional figure types necessitates additional calculations. However, in all patterns the computer can calculate at speeds over 20 fps.

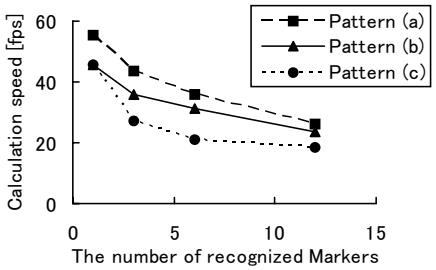


Figure 15: Relation between calculation speed and the number of recognized markers.

8 Conclusion and Future Works

We proposed and constructed the 3D position tracking system using three seamless patterns and by experiment confirmed that users can get 3D positional data from these patterns in real time.

One of the major weaknesses of our system is that marker detection falls off at greater camera angles and distances. To make our system more robust with respect to camera angles, especially when calculating the affine transformation matrices, we plan to design future versions that use not only the positions of figures in a single marker, but also the arrangements of all of the figures in a frame. We plan to solve the distance problem by registering the Fourier descriptors of a wider variety of figures and applying learning algorithms to our image processing method.

This research is the first step toward making marker-encoded patterns that would compliment the décor of a home or business. To further develop more visually pleasing markers that will convince people to try our system in their personal spaces, it is necessary to make patterns modeled on ones currently in use in wallpapers, tiles and carpets and to research with an eye for interior design.

References

[1] R. Azuma, Y. Baillot, R. Behringer, S.Feiner, S.Julier, and B.MacIntyre, "Recent advances in augmented reality," IEEE Computer Graphics and Applications, No. 21, Vol. 6, pp. 34-47, 2001.

[2] R. Malaka and A. Zipf, "Deep map – Challenging IT Research in the Framework of a Tourist Information System," Proc. of ENTER 2000, 7th International Congress on Tourism and Communication Technologies in Tourism, pp. 15-27, 2000.

[3] P. Daehne and J. Karigiannis, " Archeoguide: System architecture of a mobile outdoor augmented reality system," Proc. ISMAR2002, pp. 263-264, 2002.

[4] R. Tenmoku, Y. Nakazato, A. Anabuki, M. Kanbara, and N. Yokoya, "Nara palace site navigator: Device-independent human navigation using a networked shared database," Proc. Int. Conf. on Virtual Systems and Multimedia, pp. 1234-1242, 2004.

[5] Indoor GPS: http://www.indoorgps.com/

[6] Polhemus Sensor: http://www.polhemus.com/

[7] M. Kourogi and T. Kurata, "Personal Positioning based on Walking Locomotion Analysis with Self-Contained Sensors and a Wearable Camera," Proc. ISMAR2003, pp. 103-112, 2003.

[8] D. Sticker and T. Kettenbach, "Real-time and Markerless Vision-Based Tracking for Outdoor Augmented Reality Applications," Proc. ISAR 2001, pp. 189-190

[9] J. W. Lee, S. You, and U. Neumann, "Tracking with Omni-directional Vision for Outdoor AR Systems," Proc. ISMAR2002, pp. 47-56, 2002.

[10] L. Vacchetti, V. Lepetit, and P. Fua, "Combining edge and texture information for real-time accurate 3d camera tracking," Proc. ISMAR2004, pp. 48-57, 2004.

[11] A. J. Davison, "Real-Time Simultaneous Localisation and Mapping with a Single Camera," Proc. ICCV'03, Vol. 2, pp. 1403-1410, 2003.

[12] R. Behringer, J. Park, and V. Sundareswaran, "Model-based Visual Tracking for Outdoor Augmented Reality Applications," Proc. ISMAR2002, pp. 277-278, 2002.

[13] G. Simon, M-O. Berger, "Pose Estimation for Planar Structures," IEEE CG & A, Vol. 22, Issue 6, pp. 46-53, 2002.

[14] H. Kato, M. Billinghurst, "Marker Tracking and HMD Calibration for a Video-based Augmentec Reality Conferencing System," Proc. IWAR '99, pp. 85-94, 1999.

[15] B. Thomas, B. Close, J. Donoghue, J. Squires, P. D. Bondi, M. Morris, and W. Piekarski, "ARQuake: An Outdoor/Indoor Augmented Reality First Person Application," Proc. ISWC2000, pp. 139-146, 2000.

[16] M. Kalkusch, T. Lidy, M. Knapp, G. Reitmayr, H. Kaufmann, and D. Schmalstieg, "Structured Visual Markers for Indoor Pathfinding," Proc. ART02, 2002.

[17] G. Baratoff, A. Neubeck and H. Regenbrecht, "Interactive multi-marker calibration for augmented reality applications," Proc. ISMAR2002, pp. 107-116, 2002.

[18] L. Naimark and E. Foxlin, "Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker," Proc. ISMAR2002, pp. 27-36, 2002.

[19] S. G. Rao, S. J. Schmugge, L.F. Hodges, "Vision-based System for Head Pose Tracking in Indoor Immersive Environments," Proc. ICAT 2004, pp. 156-161, 2004.

[20] Y. Nakazato, M. Kanbara and N. Yokoya, "Localization of wearable users using invisible retro-reflective markers and an IR camera," Proc. SPIE Electronic Imaging, Vol.5664, pp. 563-570, 2005

[21] Anoto pen: http://www.anoto.com/

[22] http://www.kmfactory.jp/

[23] http://interiorlife-style.com/yuka/index.html

[24] Y. Uesaka, "A New Fourier Descriptor Applicable to Open Curves," Transaction of IEICE Vol. J67-A, No. 3, pp. 166-173, 1984

[25] FFTW: http://www.fftw.org/